

from the patient?

- c. Should the result be reported to the clinician and to the chart with some kind of qualification attached?
- d. Is there a quality control problem in the laboratory which requires immediate action?
- e. Is there a breakdown in the clinical procedure (ordering, specimen collection, etc.) which requires immediate action?

Phase II. There are a number of clinical but relatively elementary considerations which may be taken into account within the laboratory -- and which certainly should be taken into account by the knowledge-based system we propose. Examples are:

1. Logic permitting evaluation of test results taking into account basic information about the patient, i.e., age, race, sex, and ward location.
2. Logic permitting evaluation of test results taking into account previous test results in the same patient.

These pieces of information are often of critical importance in evaluating the credibility or significance of laboratory reports. Normal ranges, for example, vary for some tests with age, race, and sex. Previous results on a patient, to take another example, may be the first clue to a mismarked specimen: the blood-from-the-wrong-patient blunder which is so fundamental a problem for all laboratories.

3. Logic permitting evaluation of test results taking into account the general nature of the putative diagnosis (e.g., admitting diagnosis or treatment regimen).

It should be noted here that we are not proposing that the system permit or encourage that clinical knowledge of the patient influence the test result, but only the interpretation of the result and the handling of the specimen. A general diagnosis or even a treatment regimen can greatly influence these matters. Plasma specimens from patients on oral anti-coagulants, for example, usually should not yield normal prothrombin times; indeed for these patients, "normal" is abnormal and dangerous. The implication here is for interpretation of the result, and when to report an "abnormality" through the stat or emergency systems. Similarly, patients with leukemias, especially under chemotherapy, often have remarkably elevated uric acids which have nothing to do with the usual reasons for hyperuricacidemia.

The issues which are relevant at the patient or the clinician's level hinge upon matters of test interpretation, the possibility of needing to order further tests, the possibility of new diagnoses. There is obviously an immense amount of logic which concerns laboratory test interpretation in the context of all of the possible clinical diagnoses and management problems. We are not proposing to include this mountain of knowledge, which really pertains more reasonably to programs such as Myer's INTERNIST System.

We propose to stop with knowledge which might reasonably be construed to represent the conversation of the laboratory director with the patient's clinical physician. It is difficult to specify precisely this cut-off at the stage when we are only proposing the system. The best indication of our intent might be provided by an example.

It frequently happens that the lab director and a clinical hematologist will discuss a set of lab findings for a patient (with or without the question of errors in the findings) up to the point at which it is clear that the findings support the interpretation "iron deficiency anemia". This stage of reasoning represents a kind of intermediate between findings and diagnosis which AI systems sometimes call a concept. The semantic network system of Kulikowski, Amarel and Weiss, for instance, has such "concepts" within its logic. From the point of view of the logic we propose to write, this interpretation would be a proper termination, wholly supported by lab findings but requiring more clinical information about the patient than is obtainable from such paper systems as lab requisitions. The cause of the iron deficiency anemia would remain for another system to take up.

There are a host of such intermediate pathophysiological concepts which constitute a kind of proper frontier between clinical lab reasoning and more purely clinical reasoning. In practical terms, the resolution frequently is reached either by a telephone conversation between the lab director and the clinical physician, or by personal contact on such an occasion as rounds. We are not eager to automate the personal contact, although time does not permit enough of these discussions to occur; we would like to automate at least the decision to make the telephone call or appointment.

Most test results, even batteries of results do not permit an interpretation at the laboratory level. In some cases, we feel the logic could take us further. The most extreme case and the most complete logic we feel would end with a tentative pathophysiological concept (such as anemia) and in selected important cases a decision on the part of the computer system to recommend the lab director call the clinician. Because of the limitations of

time, this is not a minor decision. Only the most important cases should be selected for such conferences, whether telephone or in person. A system with full and explicit logic should form a good basis for such a decision. Furthermore, previous experience has shown us that even our non-AI current lab monitoring systems must bring together all pertinent (available) information about a patient before bringing the abnormal report to the attention of the user. This simple assembling of data aids current decision making; we anticipate that assembly based on a more extensive logic will prime a clinically useful discussion.

Phase III. Logic relevant to hospital function primarily concerns institutional patterns. This includes changes in laboratory patterns, timeliness of reporting, distribution of costs among services and patients, and examination of interactions between procedures. For example, do screening batteries including such tests as LDH's result in an inappropriate number of repeat kinetic enzyme studies? These matters are derivative measures of institutional function which are the natural by-products of semantic understanding of the laboratory transactions. They would not be examined until after the more fundamental logic in Steps I and II had been dealt with.

Phase IV. Logic which links to considerations outside the hospital environment.

It is difficult to detail these linkages ab initio. They are made up potentially of at least two separate concerns: derivation of facts of general scientific interest; and the provision of linkages to educational functions.

It must be emphasized that firm promises for such accomplishments

cannot be made. Still, one should point out some potentially important implications outside the immediate hospital realm, and should attempt to make the connections. A more or less modest scientific fact which could with luck result from the studies is the long awaited multivariate normal for application to multi-channel screening (Lezotte, 1977; Grams, 1977).

Building of instructional systems is beyond the scope of the present proposal, but provision of the connections is an inherent part of our plan. Good AI systems are (partly) characterized by their ability to defend their decisions. That is, a classification or advice provided from such an automated system can be challenged, and it can be expected the the system can recapitulate the rules or criteria which produced its conclusion. It is precisely this ability which should allow potential users outside the laboratory to benefit directly from the existence of such a knowledge-based system. We would hope to allow for this educational by-product usage by providing suitable means to challenge and converse with the system.

System building

We have given thought to the architecture of the proposed system. It should be emphasized that this project is a long term development in an area of fundamental importance to medicine: namely, the knowledge which surrounds clinical laboratory testing. We feel that there exists an adequate base of expertise in this field at the University of Missouri, acknowledging of course that we would utilize the full resources of the published literature and that the knowledge and logic of the system would be subjected to outside review by consultants as each major step was taken. We do not, however, have an adequate experience in work in artificial intelligence techniques per se to undertake the project alone. It is clear that this competence exists in the group at Stanford. We feel we have a sufficiently good working relationship with Professor Feigenbaum and his colleagues that a joint development will be successfully concluded.

The form of the actual computer representation has not been selected. Our lab systems have used table driven assembly code for years. The HCTC is collaborating with clinicians at UMC and computer scientists at Rutgers to create a rule-based rheumatology consultant. We wish to explore with Dr. Feigenbaum the possible appropriateness of the imputational "blackboard" of the Hearsay system.

The knowledge-based system to incorporate clinical laboratory expertise will be built on the SUMEX machine via the existing time-sharing network. We have used terminal connections to SUMEX for five years in connection with operation of the AIM network, the SUMEX Executive Committee, and smaller experimental projects. The communications are sufficient to support development of such

a system. At the same time, we recognize that it is inappropriate (and probably impossible) for the SUMEX computer complex in California to support a real-time service activity in Missouri. Fortunately this is not necessary. Testing of the model in its sequential versions against actual lab data in batches or benchmark sets can easily be done on a periodic basis. This will not be a problem. Even the status of the quality control results can be accessed and included in the model's operation in this fashion. Since all transactions are recorded, one can accurately recreate "real time" for any moment.

The issue of implementation of the full model in a real laboratory setting is a separate problem. The system has not yet been built, so we can't say what kind of computer would be needed to run it. If we are correct in assuming, like other systems, that a part of a PDP-10 is capable of running the model, then it is not unreasonable to expect our laboratories to acquire this level of computer support. The current lab systems are using a combination of two PDP-12's, an IBM System 7, substantial services of an IBM 370/158 (which is being replaced by an Amdahl machine), and several microprocessors, including M6800's and LSI-11's. All this does not add up to an AI machine, but we don't want it to yet. There is a commitment to having computing gear at UMC, and in most large clinical laboratories. At the same time, one must acknowledge that the five year duration of the project will doubtless see a continued reduction in the cost of computing gear, as well as a continuation of the advances in hardware which will have made AI techniques more realistic in the past. Machines equivalent to DEC PDP-10's may well come to be offered for small amounts of money in microforms. This kind of breakthrough is not necessary in order for us to move over to an AI-based system. What is necessary is that

the system work well and be able to keep up with the changes in laboratory procedures which have plagued and almost destroyed previous systems. Our institution is currently supporting six full time programmers in a vain attempt to keep rigid old programming systems current with methodological and administrative changes. If the AI techniques succeed in producing a competent flexible software system, we feel that ongoing personnel savings will offset even large one-time hardware costs.

While the major model system is being built, we will naturally implement as improvements whatever parts of the logic are reasonable and feasible on the existing hardware. This is not difficult to imagine, because the current system is somewhat distributed already. It is through this means that we would expect to identify and hopefully to achieve cost savings and quality improvements. We assume that the major advances would come through implementation of the full new system. These should be calculated ahead of time. If the savings and improvements are "there", the project will have been successful and the system will be implemented as a whole at UMC and elsewhere.

Concepts to be included

There are certain general concepts which are suffused throughout all elements of laboratory practice. These will necessarily be incorporated in all phases of the proposed development.

These concepts include the following:

1. Statistical significance of testing, including sensitivity - specificity of tests. This orientation is inherent in lab work. Recent reports (Casscells, Schoenberger Graboys, 1978; Ransohoff and Feinstein, 1978) indicate that it is not well understood by the clinical users of laboratory services.
2. Related to this idea is the concept of normal, which is very much dependent upon each particular laboratory, and even upon specific methodologies. The knowledge of normal ranges regarding the methodology and regarding age, sex, race, and special circumstances of the test population must be firmly associated in the system with each test specification. The system must be able to defend its interpretations, and hence to inform the user of the laboratory's assumptions and adjustments to methodology.
3. The concept that automatic error detection is the essential first step before interpretation of results is attempted, and that the attempt at error detection must be vigorous. With the present systems we are able by careful after-the-fact daily checking to recognize and correct errors in data which have passed through the computer checks and have actually been reported to the patient's chart. Two and one half percent of results are in error. Of these 0.5% (In retrospect) actually represent

true technician or technologist methodological errors. The remainder are a very mixed bag of clerical and administrative errors. Our performance (which is probably good compared with many wholly manual or semi-automated labs) is the result of incorporating extensive computer editing of the data. We long ago, for example, incorporated self-check digit identification for patient and specimen numbers, since we had shown that this category alone accounted for half the errors detected by an earlier system (Lindberg, Schroeder, Rowland, Saathoff, 1969).

Additional empirical methods of pattern recognition have been developed for error deletion, and will be incorporated in the proposed system. These include analysis of electrolyte patterns, creatinine and others (Lindberg, 1968).

The current daily Abnormal Value Rounds in the laboratories will provide an ideal work setting for the model development and testing. Presently lab reports are transmitted by and reviewed by the several computer systems. Special cases, according to adaptive algorithms, are selected by the systems for review daily by the chairman of the Department of Pathology, Dr. Townsend, and his residents and staff. They currently accept or reject the computer judgments based on their own internalized judgments and upon additional data about the patients which is obtained by going to see the patient and/or the chart. It is this logic which should be represented in the new programs.

4. Multi-step testing is a practice which has been common to labs for decades. The logic is not always made explicit to the user, and we feel there is an advantage in doing so. The classic example is the serological test for syphilis. Formerly, laboratories did a VDRL (for sensitivity), followed in the positive cases by a Mazzini (for specificity). Currently these have been replaced by the rapid plasma reagin test and the fluorescent treponema antigen test. The same practice is followed (appropriately) with many clinical enzyme tests such as CPK and LDH, their kinetic counterparts and their iso-enzyme extensions. Even more dramatic is the multi-step or branching tree logic which is used by coagulation laboratories and the special immunology laboratories. The questions to be addressed by the system include: what test should be done first? What is available locally? What subsequent test to do, dependent upon what initial results? What statistical significance do the results have? What further testing could be done? If this involves a remote referral lab, how is the service obtained?

Essentially, this logic is quite subject matter dependent. It is specific to the limited domains, but because of this, also quite synonymous with expert behavior.

III.D. Significance

The significance of a successful outcome would be:

1. Advances in basic knowledge representation techniques
2. Formal and public representation of a major field of medical expertise which will be of interest to all fields of medicine, health care, and information science.
3. Advances in techniques for remote collaboration on information system development. That is, we would be much further along on knowing how to share rare computational facilities and unique computer science competence with a broader, perhaps even national, medical community.
4. Improved understanding of evaluation of advanced health care technology.

The significance of a less than complete success would be lessened. Undoubtedly some of the representation and testing would be accomplished, since we will commence with the easiest part. If one's success were limited to this, the results would be of real importance but of interest primarily to laboratorians and computer scientists. These are an important part of the audience, but not the only ones we see for the complete system. The "downside risk", in other words, is minimal.

III.E. Facilities available

The Health Care Technology Center can house the computer component of the project at the University of Missouri-Columbia. Space is available in a modern office building. The Center provides library facilities, computer laboratory facilities, telecommunication, etc. The Department of Pathology will be providing access to the working laboratories as required. These include Hematology, Chemistry, Microbiology, Clinical Microscopy, Coagulation, Immunology and Anatomical Pathology services for the University Hospital (440 beds), a similar arrangement for the adjacent Harry S Truman Memorial Veterans Medical Center (480 beds), the Mid-Missouri Mental Health Center (175 beds), and Rusk Rehabilitation Center (100 beds). The combined laboratories process 2,100,053 procedures a year.

Computer hardware per se includes 6 DEC LSI-11's; 3 M6800 systems; 2 DEC PDP-12's (tapes, disks, terminals); DEC PDP11/34; IBM System 7; and multiple direct connections to the University Network IBM 370/158 and 370/168 (both to be replaced by Amdahl gear).

The members of the Health Care Technology Center include 45 faculty from 14 University departments in 6 schools of the Columbia campus.

The professional staff of the Department of Pathology includes 29 faculty and 20 residents and fellows. Only a subset of the faculty are planned as active members of this project team, but all are interested in the success of the venture and all are available as needed for help on specific knowledge areas within their own subspecialties.

III.F. Collaborative arrangements

The system would be developed jointly with members of Computer Science at Stanford and the Health Care Technology Center at the University of Missouri-Columbia. Computer support for the model system would be provided by the SUMEX computer facility. This is an NIH supported national resource. Use of local computers at UMC for data gathering, analysis, test implementation would be provided free of charge. An exception is minor maintenance charges for HCTC equipment. Telecommunications for approved projects are provided by the SUMEX contract with TYMNET and ARPANET. Access to Net nodes is provided by UMC WATS lines. In addition, the project would budget funds to provide for frequent travel between the two schools.

Results of the project are to be published.

Stanford University is viewed as the primary submitter of the proposed program project, with the University of Missouri-Columbia supporting the application and taking responsibility for the Laboratory Expert Project. Doctor Feigenbaum is the Principal Investigator for the program project. Doctor Lindberg is viewed as Director of the Laboratory Project.

PROJECT 3: REFERENCES

1. Shortliffe, E.H., Axline, S. G., Buchanan, B.G., Merigan, T.C. and Cohen, S. N., "An Artificial Intelligence Program to Advise Physicians regarding Antimicrobial Therapy". Computers and Biomedical Research, 6 (1973):1-17.
2. Weiss, S., Kulikowski, C. A., Safir, A. "Glaucoma Consultation by Computer". Computers in Biology and Medicine, 8 (1978): 25-40.
3. Pauker, S. G., Gorry, G. A., Kassirer, J. P., Schwartz, W. B. "Towards the Simulation of Clinical Cognition: Taking a Present Illness by Computer". American Journal of Medicine, 60, (June, 1976): 981-996.
4. Lawrence, S. V. "Internist: Computer Program Expressing Clinical Experience and Judgment of a Master Internist Constitutes a Unique Resource". Forum on Medicine (April 1978): 44-47.
5. Hicks, G.P., Evenson, M.A., Gieschen, M. M., Larson, F.C. "On Line Data Acquisition in the Clinical Laboratory". Computers in Biomedical Research Vol. III (Stacey and Waxman) New York: Academic Press, 1969, pp. 15-53.
6. Lindberg, D. A. B.: "Collection, Evaluation and Transmission of Hospital Laboratory Data". Proceedings 7th IBM Medical Symposium (1965): White Plains, New York, IBM, 1965.
7. O'Kane, K. C., Haluska, E. A. "Perspectives in Clinical Computing". In Advances in Computers, 16 (1977): Academic Press, 161.
8. Lezotte, D. C. "A Multivariate Laboratory Data Analysis System: Introduction". Journal of Medical Systems, 1, No. 3 (1977): 293-98.
9. Grams, R. R. "Progress Toward a Second Generation Laboratory Information System (LIS)". Journal of Medical Systems, (1) No. 3, (1977):263-74.
10. Casscells, W., Schoenberger, A., Graboys, T., "Interpretation by Physicians of Clinical Laboratory Results". New England Journal of Medicine 299, No.18 (November 1978): 999-1001.
11. Ransohoff, D. F., Feinstein A. R., "Problems of Spectrum and Bias in Evaluating the Efficacy of Diagnostic Tests". New England Journal of Medicine 299, No. 17 (October 26, 1978): 926-30.
12. Lindberg, D.A.B., Schroeder, J.J., Jr., Rowland, L.R., Saathoff, J., "Experience with a Computer Laboratory Data System". In Strandjord, J. (ed), Multiple Laboratory Screening. Academic Press, New York, 1969, 245-55.

The undersigned agrees to accept responsibility for the scientific and technical conduct of the research project and for provision of required progress reports if a grant is awarded as the result of this application.

1/22/78
Date

Donald A. Smith
Principal Investigator

IV. CORE RESEARCH

IV.A. Objectives of Research

The long term goal of artificial intelligence research at the Heuristic Programming Project (HPP) is to understand and build knowledge-based "intelligent agent" programs. Over the past decade we have studied such systems in the context of scientific and medical applications where human expertise for solving the problems was evident and where the difficulty of the problem seemed to lie just outside the boundaries of current AI methods. Because of the complexity of the applications, a significant part of the effort has been to make the expert knowledge of the problem explicit and to represent it appropriately in a knowledge base. This perspective has focussed attention on four areas for research:

- (1) Representation — designing the symbolic structures for modeling the knowledge about a problem. Presently this phase is carried out by the system builders; we intend to codify the knowledge used to make such decisions, both as an aid to the system builders and ultimately to enable the programs themselves to choose appropriate representations.
- (2) Reasoning — modeling the appropriate inference mechanisms for a problem and building systems that incorporate those models.
- (3) Knowledge acquisition — designing systems that acquire knowledge by communication with human experts.
- (4) Multiple uses of knowledge — designing systems that use the symbolic representation of the domain knowledge for additional purposes such as consensus building (accommodating conflicting advice from experts whose competence may be equal but whose "styles" vary), tutoring of human students by employing the knowledge base (both the information it contains and the way it is organized), and explanation (constructing a chain of rules which satisfactorily rationalize the system's behavior to an observer).

IV.B. Background and Rationale

Artificial intelligence research at the Heuristic Programming Project has utilized medical and scientific problems to focus the research effort. For many different applications over the last decade this has led to a cycle of research as follows:

1. Form a collaboration with a scientist to work on a specific problem in a challenging and interesting area.
2. Propose a method for representing and manipulating the domain knowledge. This involves acquiring both formal and informal knowledge and developing a knowledge-based system that reasons with that knowledge.
3. Test the system. In this phase the method is pushed to its limits. The relationship between the design and the performance of the system is used as the basis for future development.

Both success and failure of a system can lead to further research steps. When a system fails to solve a problem, the seeds for further research can sometimes be found in the reasons for failure. On the other hand, when a knowledge-based system is successful, the desire to use it effectively uncovers a number of additional needs. Thus, many of the topics of artificial intelligence — such as the ability of a program to acquire knowledge, or to explain its reasoning, or to manage updates in a knowledge base — have grown out of programs that were at first successful only at problem solving. From this experience has come not only a set of approaches to building intelligent systems, but also a broader understanding of what intelligent systems should be like.

The following sections discuss the background information about each of our major research areas. We will outline the progress that has been made on this topic and identify the major technological tools. Then in Section IV.C. we will discuss our perception of the outstanding research issues and how we plan to approach them.

IV.B.1. Representation

One of the trends in our work has been to develop general purpose approaches for representing a broad range of knowledge in a knowledge base. This is illustrated by the Unit Package that has been developed for the MOLGEN project([40],[53]) for experiment planning in molecular genetics. In the figure below are two units from a MOLGEN knowledge base. The first unit represents the restriction-enzyme EcoRI; the second unit represents a problem-solving goal for an experiment.

```

NAME:                ECOR1
SITE-TYPE:           STICKY-HEXA
3'-END:              OH
5'-END:              P
MODE:                NON-PRECESSIVE
MOLWT:               28500
SUBSTRATE:           DNA
RECOGNITION-SITE:

                    1  2  3  4  5  6  7  8
                    G  A  A  T  T  C
                    _____
                    C  T  T  A  A  G
                    16 15 14 13 12 11 10 9

```

```

NAME:                LAB-GOAL-1
STATE:               A CULTURE with
                    ORGANISMS = A BACTERIUM with
                    EXOSOMES = A VECTOR with
                    GENES = RAT-INSULIN
CONDS:               (PURE? ORGANISMS CULTURE)

```

The usual way of using the Unit Package is to define general knowledge before specific knowledge. For example, general knowledge about enzyme, nuclease, and restriction enzymes would be entered before the specific knowledge about a particular restriction enzyme like EcoRI. The Unit Package is designed to encourage the use of description, such as the description of a culture in the second unit above. These descriptions are used for checking new information as it is entered and for pattern-matching operations that are part of a reasoning step. Reference [52] describes the Unit Package and compares it to other work on representation.

The examples above have illustrated the representation of "object-centered" or "noun-like" knowledge. Every reasoning program also contains a representation of the inferential

knowledge. In the first version of the DENDRAL program, this kind of knowledge was represented as a program. This choice of representation had the consequence that a chemist could not enter new knowledge into the program (because he could not be presumed to be an expert programmer). Also, since the program structures were not understandable by the program itself, facilities for explanation of DENDRAL's reasoning had to be built into each part of the program. In the MYCIN program [51], developed more recently, the inferential knowledge was moved out of the program and into a knowledge base represented as production rules. This representation, because it was closer to the experts' representation than DENDRAL code was, allowed us to develop programs that could acquire rules from physicians. It also allowed the system to generate its own explanations by examining the rules it had used. Production rules illustrate many of the themes which run through our work on representation.

- (1) Explicitness — Knowledge is encoded in a knowledge base and not just in programs. (For example, production rules are used to make inferential knowledge explicit.) The distinction between knowledge being in a program or in a knowledge base is a crucial one, for our purposes. Information encoded as a program can be run, and initially coded, more easily and quickly. However, as the program grows, it becomes more and more difficult to add new knowledge: its relationships to all the other knowledge must be considered and programmed explicitly. The latter method, storing knowledge in a separate data structure, a "knowledge base", enables the pieces of knowledge to be accessed and manipulated just like data. While their use, their running, may be somewhat slower, the system builder can now enter data in modular fashion, without much concern for the rest of the items in the knowledge base. He can give the system the knowledge it needs to reason about its own knowledge base.
- (2) Modularity — Knowledge is encoded in independent "chunks" as far as possible. (Production rules can be added or deleted from a knowledge base to change its problem-solving behavior.) The concepts chosen to represent the chunks of knowledge are those which are natural and useful to a domain expert. This is useful both if the expert is to input rules directly, and if he is to be convinced by the system's explanation of its behavior.
- (3) Uniformity — Knowledge is represented so that it can be manipulated by general purpose programs. (Production rules and frames are two of the uniform methods for which we have general purpose processing routines.)

Our perception of the outstanding research issues in representation is discussed in Section IV.C.1.. As can be seen from the examples above, how knowledge is to be used is important in determining how it should be represented. With more uses for knowledge — explanation, tutoring, problem-solving — come more constraints on its representation.

IV.B.2. Reasoning

The first step in creating a problem-solving system is to develop and test a method for reasoning. In the DENDRAL program[11] for inferring chemical structures from mass spectrometry data, the reasoning framework that we tested was called the Generate-and-test paradigm. This consisted of (1) an exhaustive generator of all possible solutions (chemical structures) and (2) a set of pruning rules which used the mass spectrometry data to eliminate inconsistent answers. One of the issues that became relevant in studying this reasoning framework is the combination of possibly contradictory evidence. Data in many problems is incomplete and errorful; there is seldom a perfect match between an internal model and empirical data. Even if DENDRAL had a perfect model of how mass spectrometry data corresponds to chemical structures, the data from any particular run of a mass spectrometer are erroneous with respect to both extraneous and missing data. In DENDRAL, an overall domain-specific matching function was used which reflected a priori probabilities of errors in the data. Recently we have reexamined this problem in the context of the GAL program[53] which solves an analogous problem from molecular genetics.

For the MYCIN program we used backwards-chaining as a reasoning framework. This method develops a line of reasoning by chaining together MYCIN's inference rules (production rules) backwards from the goal of making the diagnosis towards the available evidence. This particular reasoning framework has proved especially convenient for developing computer explanations of the program's reasoning. To deal with imperfect evidence and inexact rules of inference, a mathematical model of certainty based on numeric "certainty factors" was developed. This constitutes a model of "plausible reasoning". In order to test the MYCIN approach in other domains, a domain independent package, EMYCIN (for "Essential MYCIN") has been created and is being utilized in other applications discussed elsewhere in this proposal.

When MYCIN is chaining back through its inference rules and discovers a need for information that cannot be inferred, it stops and asks for it. This approach is appropriate only when

there is a way of supplying data as needed by the reasoning program. For some applications, such as signal interpretation, it is better for the program to make use of whatever it knows, because there is little chance that specific items of information can be supplied on demand. Further limitations of a simple backwards-chaining model are (1) it is unidirectional, hence cannot mix top-down and bottom-up processing and (2) it is exhaustive, hence less efficient than approaches that reason hierarchically by working with abstractions.

An alternative reasoning model which does not have these limitations is the "cooperating knowledge sources" model developed for the HEARSAYII [20] system and incorporated in our AGE-I program. This model consists of (1) the "blackboard", a global data structure which holds the system's hypotheses, and (2) a set of "knowledge sources" (KSs) which contain the inference rules for the system. Because of gaps in the theory and implementation of the individual KSs and noise in the data, the KSs are individually incomplete and errorful. A version of the "hypothesize and test" paradigm is used which emphasizes cooperation (to help overcome incompleteness in both knowledge and data) and cross-checking (to help correct errors). During the hypothesize part of the cycle, a KS can add a hypothesis to the blackboard; during the test part of the cycle, a KS can change the rating of a hypothesis in the blackboard. This process terminates when a consistent hypothesis is generated satisfying the requirements of the overall solution or when knowledge is exhausted. The power of the blackboard — over, say, a uniform QA4 assertional net — is its structure: it is n-dimensional, where the dimensions have some meaning (time, level of abstractness, geographic location, etc.). Hence each rule can know what part(s) of the blackboard to monitor, and each hypothesis is carefully placed at a meaningful spot on the blackboard. This is a simple but powerful type of analogic modelling of the domain.

Two research programs based on this paradigm have been developed by our group [43]. One is the CRYSTALIS program for interpreting x-ray crystallography data and the other is a military signal interpretation program. In these programs the HEARSAY model was extended by (1) extending the blackboard to allow for several independent hierarchical relationships among data and hypotheses and (2) extending the control structure.

In each of the examples above, our study of reasoning methods always starts in the context of a problem in a scientific or medical domain. We then generalize the method and package it for further testing in other domains. When a framework for reasoning works well enough, research on other artificial intelligence topics, such as explanation or knowledge

acquisition, often follows. Our perception of open research issues in reasoning methods is discussed in Section IV.C.2..

IV.B.3. Knowledge Acquisition and Management

One characteristic of the domain problems we have studied is their requirement for a substantial amount of domain expertise. Goldstein addressed this point in [26]:

Today there has been a shift in paradigm. The fundamental problem of understanding intelligence is not the identification of a few powerful techniques, but rather the question of how to represent large amounts of knowledge in a fashion that permits their effective use and interaction. This shift is based on a decade of experience with programs that relied on uniform search or logistic techniques that proved to be hopelessly inefficient when faced with complex problems in large knowledge spaces.

The relevant problem solving knowledge includes much formal and informal expertise of the domain expert; it also includes many mundane facts and figures that make up the elementary knowledge of the domain. Before a computer system can solve problems in the domain, this information must be transferred from the expert to the computer.

Over the last decade, there has been some encouraging progress along this dimension. In DENDRAL, the rules of inference about mass spectrometry had to be put in machine form, but knowledge acquisition by the program from the chemist was beyond our technology. Knowledge was added by a painstaking process in which a computer scientist together with a chemist learned each other's terminology and then wrote down the chemical rules for the simplest kinds of chemical compounds. Then the computer scientist entered the rules into the computer and tested them and reported the results back to the chemist. The reward for this effort over several years was a program with expert-level performance.

It is interesting to compare the knowledge acquisition effort of the DENDRAL program with that of a more recent program

— PUFF, the system for diagnosing pulmonary function disorder. In contrast with DENDRAL, PUFF was created in less than 50 hours of interaction with experts at PMC and with less than 10 man-weeks of effort by the knowledge engineers. Part of this tremendous difference in development time is due to the fact that the domain of pulmonary function is much simpler than mass spectrometry. However, the main reason that the development was so rapid is that PUFF was built with the aid of an interactive knowledge engineering tool, EMYCIN. When knowledge engineers at the Heuristic Programming Project started the PUFF project, they already had a reasoning framework in which to fit the problem and an "English-like" language for expressing the diagnostic rules. The facilities that make EMYCIN such a powerful tool are the direct result of the core research over the last five years on the MYCIN program.

Another dimension of progress closely related to knowledge acquisition is knowledge management, that is, management of the global structure of a knowledge base. A knowledge base is more than a set of isolated facts: its elements are related to one another. In the DENDRAL program, all of the knowledge was represented as programs and LISP data structures. If changing one part of the program meant that another part had to be changed as well, the programmer had to know that. As programs or knowledge bases get large, this kind of effort becomes substantial. A system becomes too large to maintain when no one can remember all of the interactions and every change introduces bugs. TEIRESIAS[15] extends the idea (developed initially in automatic programming research) that a system can aid substantially in identifying sources of errors and can take on some of the responsibility for making changes.

Research issues in knowledge acquisition and management are discussed in Section IV.C.3..

IV.C. Methods of Procedure

We are interested in exploring the effects of new ideas about knowledge based programming on a variety of systems to effectively test the generality of these ideas. Each of the topics in the core research area will be developed in the context of more than one example program (see discussions of Projects 1-3).

The expert systems developed at the Heuristic Programming Project over the last decade can be used as tools for the

development of the core research topics. Each of the biomedical domains has particular aspects that can be utilized in this work: the MOLGEN program for molecular genetics research has methods for representing experiment planning, the MYCIN program for infection disease diagnosis and therapy has a well developed rule set, the PUFF program for pulmonary function test interpretation has a small rule set, and the VM program for interpreting physiological measurements from the Intensive Care Unit has a knowledge base that emphasizes knowledge that changes over time.

IV.C.1. Representation

In Section IV.B.1. we traced our work from specialized representations as in the DENDRAL program to representations of more general applicability — such as our production rule and frame methodology. Today's representation systems, even the "general" ones, do not solve all of the problems that we are encountering in our research. In most science, methods which are general are also weak. There seems always to be a need to tailor aspects of a representation to particular problems. The following representation issues stand out in our work:

Time-based knowledge

Several problems which we are working on involve situations that evolve over time. In the Ventilator Management (VM) program [21], time enters as instrument data that varies over time. The program must correctly track the stages of treatment on the treatment machines. In the RX program [5] for reasoning from time-based clinical data bases, statements about disease and treatment of patients need to be adequately quantified over time. In the MYCIN [51] work, we want the system to be able to resume a consultation session about a patient and appropriately update new knowledge about the patient as treatment progresses. In the MOLGEN project [40], the experiment planning program must plan a sequence of steps. It must predict how the laboratory objects will be changed over time as the manipulations proceed. The basic issues common to these projects are (1) time-specified reference to objects and (2) tracking causal changes on objects over time. While these problems do not seem conceptually difficult, they do require extensions to the representational tools which we have available.

Grain Size in Complex Systems